WHAT IS CLAIMED IS:

1.   A method of graphics processing, said method comprising:

determining a non-depth conditional status of a fragment corresponding to a pixel;

determining that a scratchpad contains an entry mapped to the pixel; and

comparing a first value of the fragment to a value of the entry,

wherein determining a non-depth conditional status of a fragment includes determining whether incorporation of a second value of the fragment into the pixel is conditional on a non-depth criterion.

2.   The method of graphics processing according to claim 1, wherein the first value of the fragment includes a Z value.

3.   The method of graphics processing according to claim 1, wherein the second value of the fragment includes a color value.

4.   The method of graphics processing according to claim 1, wherein determining a non-depth conditional status of a fragment includes determining a current configuration of a pixel pipeline.

5.   The method of graphics processing according to claim 4, wherein determining a current configuration of a pixel pipeline includes determining a value of at least one state variable.

6.   The method of graphics processing according to claim 1, wherein determining a non-depth conditional status of a fragment includes determining whether a non-depth fragment test is enabled.

7.   The method of graphics processing according to claim 1, wherein determining a non-depth conditional status of a fragment includes determining whether an alpha test is enabled.

8.   The method of graphics processing according to claim 1, wherein said determining a non-depth conditional status of a fragment occurs before said comparing a first value of the fragment to a value of the entry.

9.   The method of graphics processing according to claim 1, wherein said determining a non-depth conditional status of a fragment occurs after said comparing a first value of the fragment to a value of the entry.

10.   The method of graphics processing according to claim 1, wherein comparing a first value of the fragment to a value of the entry includes determining whether a Z value of the fragment is less than the value of the entry.

11.   The method of graphics processing according to claim 1, further comprising overwriting the value of the entry with the first value of the fragment.

12.   The method of graphics processing according to claim 1, further comprising passing the fragment to a pixel pipeline.

13. The method of graphics processing according to claim 1, wherein determining that a scratchpad contains an entry mapped to the pixel includes determining that the entry is valid.

14. The method of graphics processing according to claim 1, wherein determining that a scratchpad contains an entry mapped to the pixel includes determining that the scratchpad contains a line of entries, the line being mapped to a block of pixels that includes the pixel.

15. The method of graphics processing according to claim 14, wherein determining that the scratchpad contains a line of entries includes determining that the line is valid.

16. The method of graphics processing according to claim 14, further comprising initializing a value of each among the line of entries to the backmost among a set of Z values.

17. The method of graphics processing according to claim 14, further comprising initializing a value of each among the line of entries to the backmost among a set of representative Z values.

18. The method of graphics processing according to claim 1, further comprising comparing the first value of the fragment to a representative Z value corresponding to the fragment.

19. The method of graphics processing according to claim 18, further comprising overwriting the representative Z value.

20.   The method of graphics processing according to claim 19, wherein determining that a scratchpad contains an entry mapped to the pixel includes determining that the scratchpad contains a line of entries, the line being mapped to a block of pixels that includes the pixel, and

wherein overwriting the representative Z value includes comparing the representative Z value with the backmost Z value of the line.

21.   The method of graphics processing according to claim 20, wherein said comparing the first value of the fragment to a representative Z value occurs before said determining a non-depth conditional status of a fragment.

22.   The method of graphics processing according to claim 1, further comprising initializing the value of the entry to an initial value.

23.   The method of graphics processing according to claim 22, wherein the initial value is a maximum Z value.

24.   The method of graphics processing according to claim 22, wherein the initial value is the backmost among a set of Z values.

25.   The method of graphics processing according to claim 22, wherein the initial value is the backmost among a set of representative Z values.

26.   The method of graphics processing according to claim 22, wherein the initial value is a representative Z value corresponding to a location to which the entry is mapped.

27.   A method of graphics processing, said method comprising:

determining a non-depth conditional status of a fragment corresponding to a pixel;

determining whether a scratchpad contains an entry mapped to the pixel; and

if the scratchpad contains an entry mapped to the pixel, comparing a first value of the fragment to a value of the entry, and otherwise altering a portion of the scratchpad,

wherein determining a non-depth conditional status of a fragment includes determining whether incorporation of a second value of the fragment into the pixel is conditional on a non-depth criterion.

28.   The method of graphics processing according to claim 27, wherein the first value of the fragment includes a Z value, and wherein the second value of the fragment includes a color value.

29.   The method of graphics processing according to claim 27, wherein determining a non-depth conditional status of a fragment includes determining whether a non-depth fragment test is enabled.

30.   The method of graphics processing according to claim 27, wherein comparing a first value of the fragment to a value of the entry includes determining whether a Z value of the fragment is less than the value of the entry.

31.   The method of graphics processing according to claim 27, further comprising initializing the value of the entry to the backmost among a set of Z values.

32.   The method of graphics processing according to claim 27, further comprising initializing the value of the entry to the backmost among a set of representative Z values.

33.   The method of graphics processing according to claim 27, wherein altering a portion of the scratchpad includes storing the first value of the fragment to the entry.

34.   The method of graphics processing according to claim 27, wherein altering a portion of the scratchpad includes mapping a line of the scratchpad to a block of pixels that includes the pixel.

35.   The method of graphics processing according to claim 34, wherein altering a portion of the scratchpad includes initializing a value of each among the line of entries to the backmost among a set of Z values.

36.   A method of graphics processing, said method comprising:

determining a non-depth conditional status of a fragment corresponding to a pixel; and

determining an occlusion status of the fragment,

wherein determining a non-depth conditional status of a fragment includes determining whether incorporation of a color value of the fragment into the pixel is conditional on a non-depth criterion.

37. The method of graphics processing according to claim 36, wherein determining a non-depth conditional status of a fragment includes determining whether a non-depth fragment test is enabled.

5   38. A graphics architecture comprising:

an early culler configured and arranged to receive a fragment corresponding to a pixel; and

a scratchpad configured and arranged to store a value of an entry mapped to the pixel,

wherein the early culler is further configured and arranged to compare a first value of 10 the fragment to the value of the entry, and

wherein the early culler is further configured and arranged to determine whether incorporation of a second value of the fragment into the pixel is conditional on a non-depth criterion.

15   39. The graphics architecture according to claim 38, wherein the early culler is configured and arranged to compare a Z value of the fragment to the value of the entry, and

wherein the early culler is configured and arranged to determine whether incorporation of a color value of the fragment into the pixel is conditional on a non-depth criterion.

20

    40. The graphics architecture according to claim 38, wherein the early culler is configured and arranged to determine whether a non-depth fragment test is enabled.

    41. The graphics architecture according to claim 38, wherein the early culler is 25 configured and arranged to determine whether a Z value of the fragment is less than the value of the entry.

42.   The graphics architecture according to claim 38, wherein the early culler is further configured and arranged to initialize the value of the entry to the backmost among a set of Z values.

43.   The graphics architecture according to claim 38, wherein the early culler is further configured and arranged to initialize the value of the entry to the backmost among a set of representative Z values.

44.   The graphics architecture according to claim 38, further comprising a pixel pipeline configured and arranged to receive the fragment from the early culler.